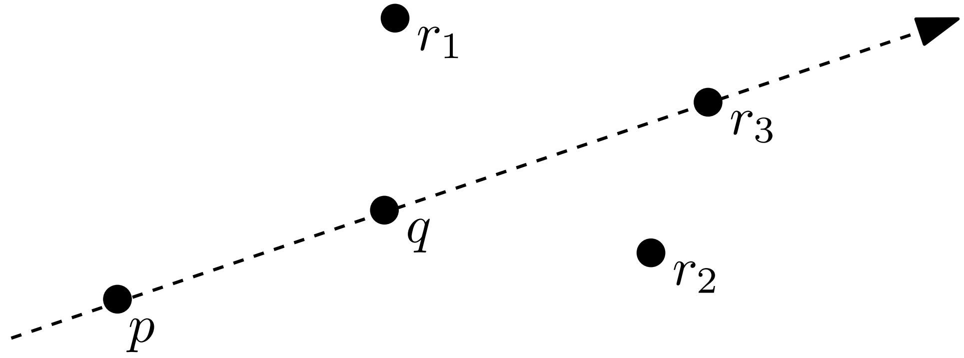
Basic tool: orientation tests

Vera Sacristán

Computational Geometry
Facultat d'Informàtica de Barcelona
Universitat Politècnica de Catalunya

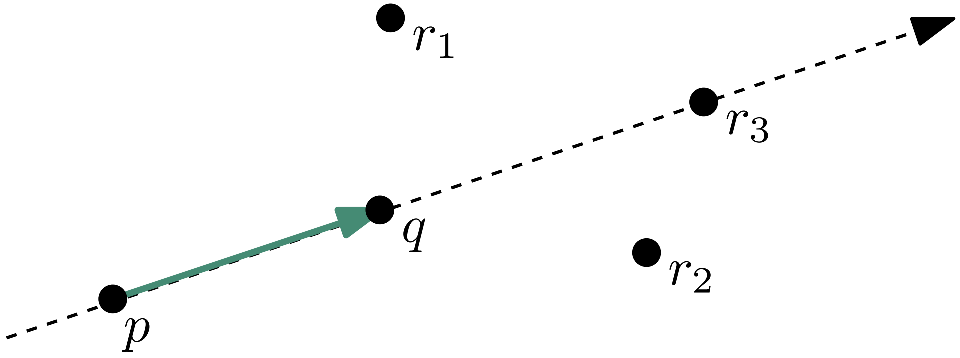
Orientation test in $\mathbb{R}^2$

Given 3 points p, q, r in the plane, efficiently and robustely decide whether r lies to the left, to the right or on the oriented line pq.



Orientation test in $\mathbb{R}^2$

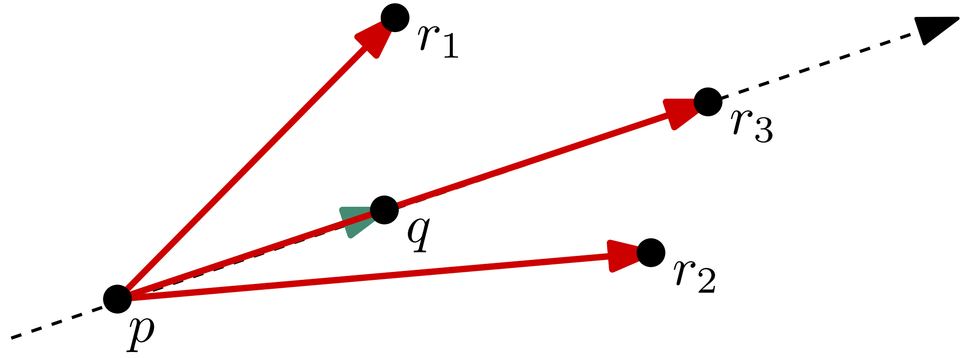
Given 3 points p, q, r in the plane, efficiently and robustely decide whether r lies to the left, to the right or on the oriented line pq.



Consider the vectors $\overrightarrow{pq}$ ...

Orientation test in $\mathbb{R}^2$

Given 3 points p, q, r in the plane, efficiently and robustely decide whether r lies to the left, to the right or on the oriented line pq.

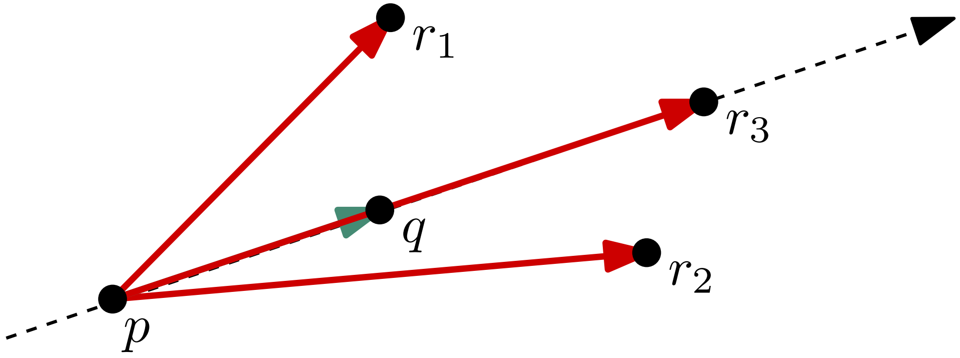


Consider the vectors $\overrightarrow{pq}$ and $\overrightarrow{pr}$.

Orientation test in $\mathbb{R}^2$

Given 3 points p, q, r in the plane, efficiently and robustely decide whether r lies to the left, to the right or on the oriented line pq.

Consider the vectors $\overrightarrow{pq}$ and $\overrightarrow{pr}$.



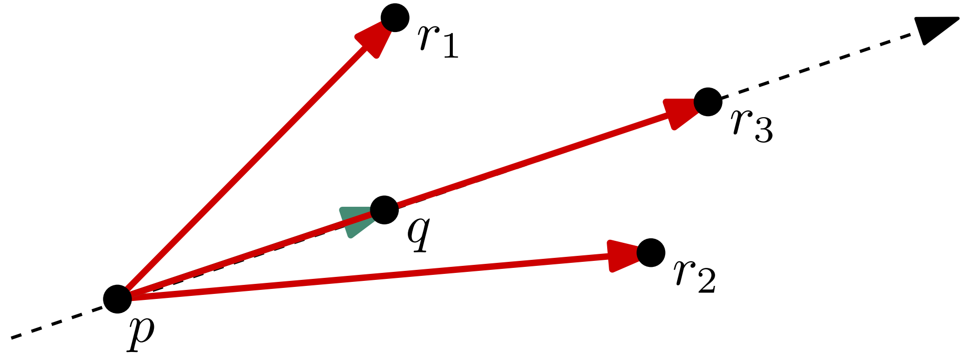
Point r lies on the line $pq \iff \begin{vmatrix} q_x - p_x & r_x - p_x \\ q_y - p_y & r_y - p_y \end{vmatrix} = 0$

Point r lies to the left of the oriented line $pq \iff \begin{vmatrix} q_x - p_x & r_x - p_x \\ q_y - p_y & r_y - p_y \end{vmatrix} > 0$

Point r lies to the right of the oriented line $pq \iff \begin{vmatrix} q_x - p_x & r_x - p_x \\ q_y - p_y & r_y - p_y \end{vmatrix} < 0$

Orientation test in $\mathbb{R}^2$

Given 3 points p, q, r in the plane, efficiently and robustely decide whether r lies to the left, to the right or on the oriented line pq.



Consider the vectors $\overrightarrow{pq}$ and $\overrightarrow{pr}$.

Point r lies on the line $pq \iff \begin{vmatrix} q_x - p_x & r_x - p_x \\ q_y - p_y & r_y - p_y \end{vmatrix} = 0$

Point r lies to the left of the oriented line $pq \iff \begin{vmatrix} q_x - p_x & r_x - p_x \\ q_y - p_y & r_y - p_y \end{vmatrix} > 0$

Point r lies to the right of the oriented line $pq \iff \begin{vmatrix} q_x - p_x & r_x - p_x \\ q_y - p_y & r_y - p_y \end{vmatrix} < 0$

Notation

$$det(p, q, r) = \begin{vmatrix} q_x - p_x & r_x - p_x \\ q_y - p_y & r_y - p_y \end{vmatrix} = \begin{vmatrix} p_x & q_x & r_x \\ p_y & q_y & r_y \\ 1 & 1 & 1 \end{vmatrix}$$

Relative position point - line

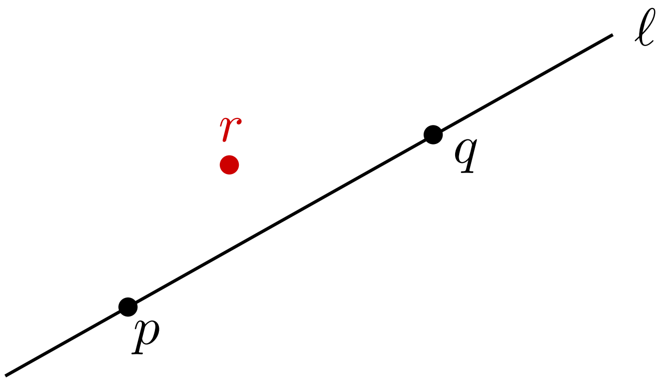
Input:

ℓ: a line (through points p and q)

r: a point

Output:

Relative position of r w.r.t. ℓ.



Relative position point - line

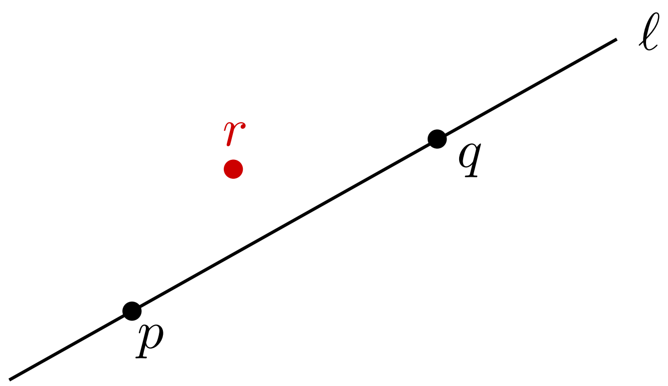
Input:

ℓ: a line (through points p and q)

r: a point

Output:

Relative position of r w.r.t. ℓ.



Intersection test line segment - line

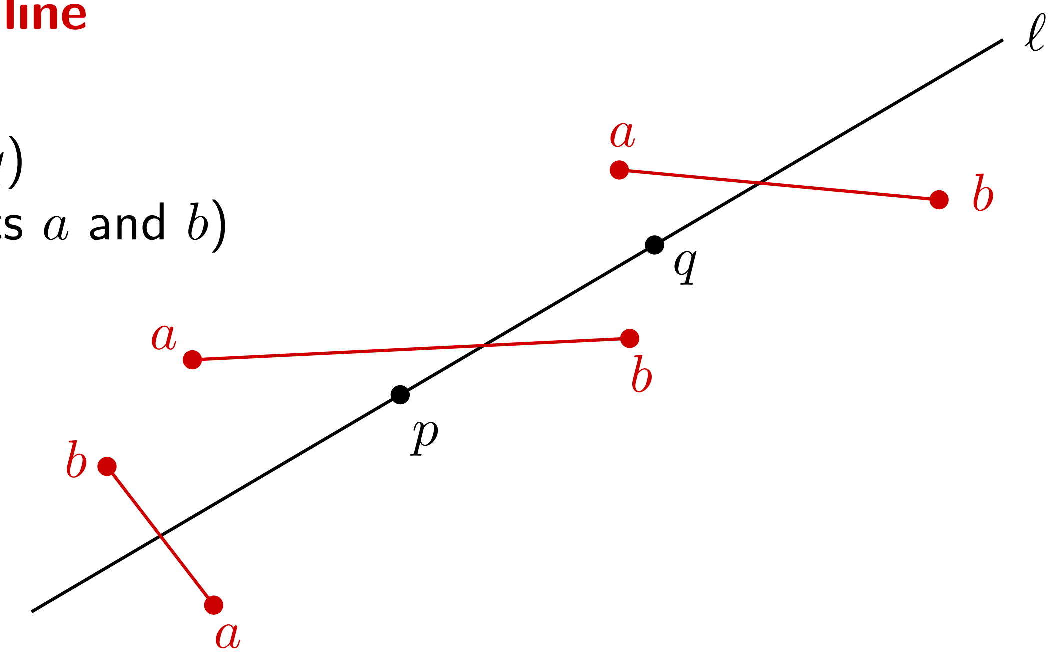
Input:

ℓ: a line (through points p and q)

s: a line segment (with endpoints a and b)

Output:

Yes/No they intersect



Intersection test line segment - halfline

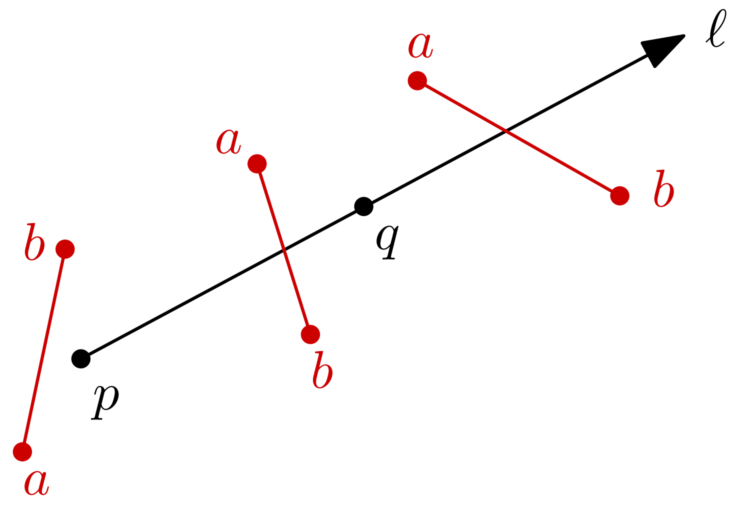
Input:

ℓ: halfline (from p through q)

s: a line segment (with endpoints a and b)

Output:

Yes/No they intersect



Intersection test line segment - halfline

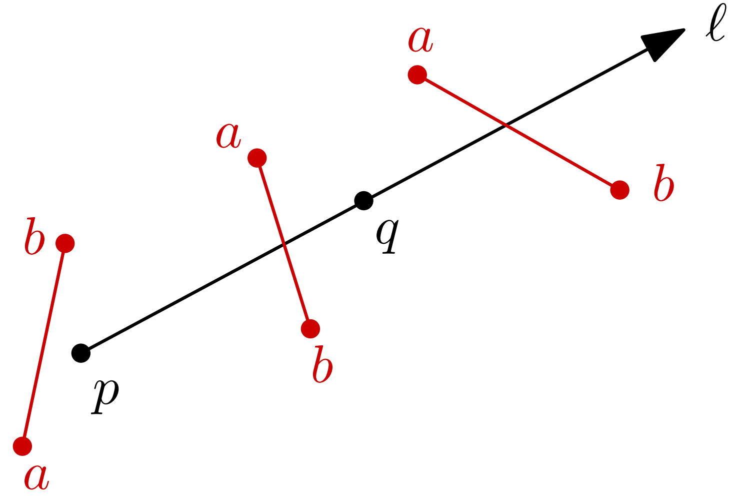
Input:

ℓ: halfline (from p through q)

s: a line segment (with endpoints a and b)

Output:

Yes/No they intersect



Intersection test line segment - line segment

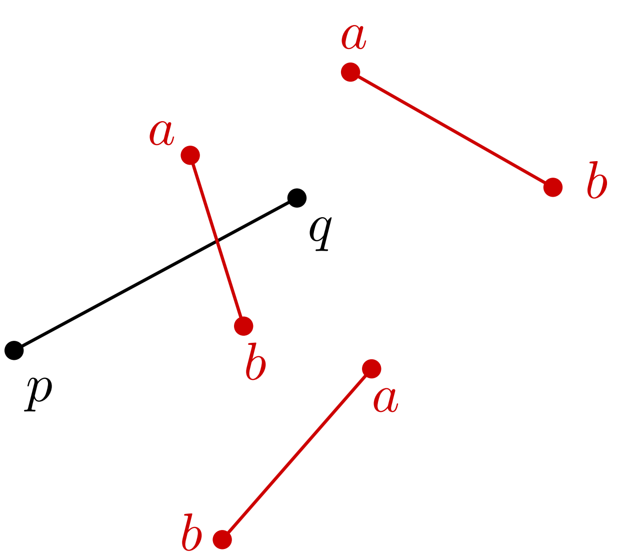
Input:

s_1: a line segment (with endpoints p and q)

s_2: a line segment (with endpoints a and b)

Output:

Yes/No they intersect



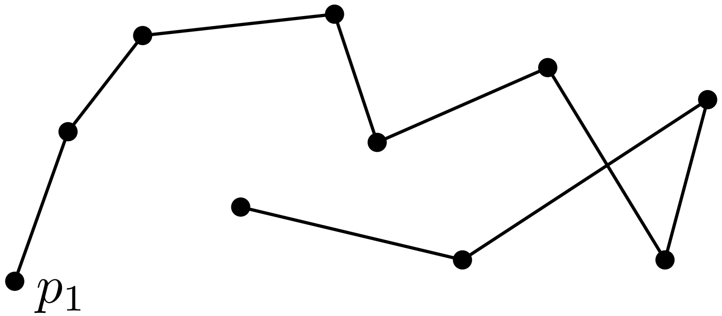
Turn orientation

Input:

A polygonal line $p_1, p_2, \ldots, p_n$

Output:

Left/right classification of its turns



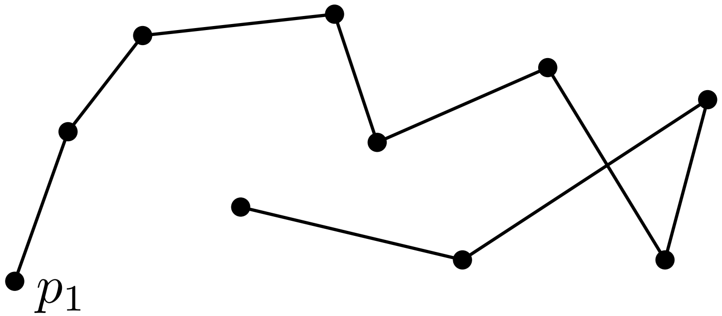
Turn orientation

Input:

A polygonal line $p_1, p_2, \ldots, p_n$

Output:

Left/right classification of its turns



Point in triangle test

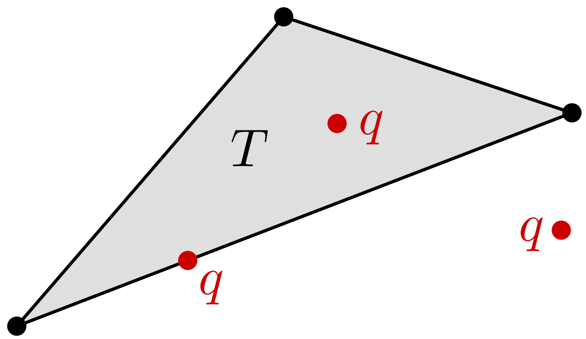
Input:

A triangle T with vertices p_1, p_2, p_3

A query point q

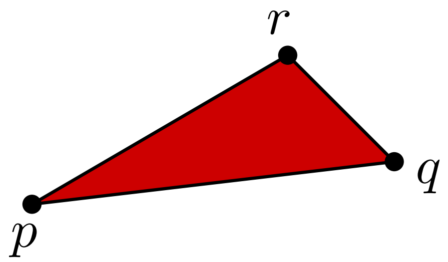
Output:

Relative position of q w.r.t. T



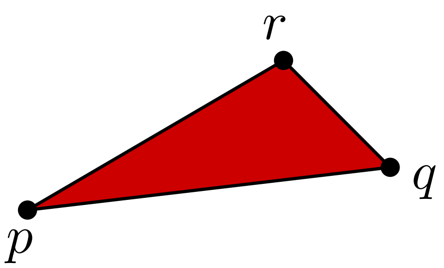
What happens in $\mathbb{R}^3$?

Oriented area of a triangle



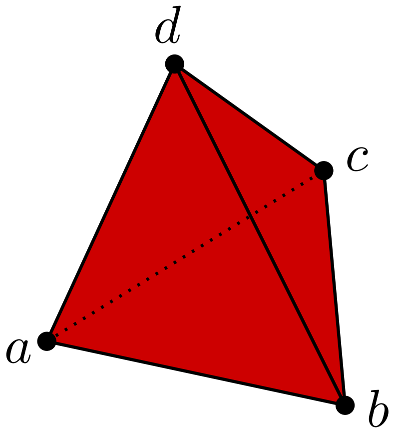
$$\text{Oriented area } (p,q,r) = \frac{1}{2}\begin{vmatrix} p_x & q_x & r_x \\ p_y & q_y & r_y \\ 1 & 1 & 1 \end{vmatrix}$$

Oriented area of a triangle



$$\text{Oriented area } (p,q,r) = \frac{1}{2}\begin{vmatrix} p_x & q_x & r_x \\ p_y & q_y & r_y \\ 1 & 1 & 1 \end{vmatrix}$$

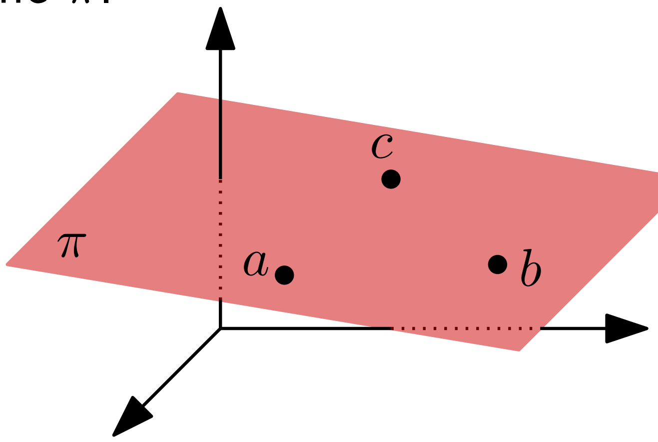
Oriented volume of a tetrahedron



$$\text{Oriented volume } (a,b,c,d) = \frac{1}{6}\begin{vmatrix} a_x & b_x & c_x & d_x \\ a_y & b_y & c_y & d_y \\ a_z & b_z & c_z & d_z \\ 1 & 1 & 1 & 1 \end{vmatrix}$$

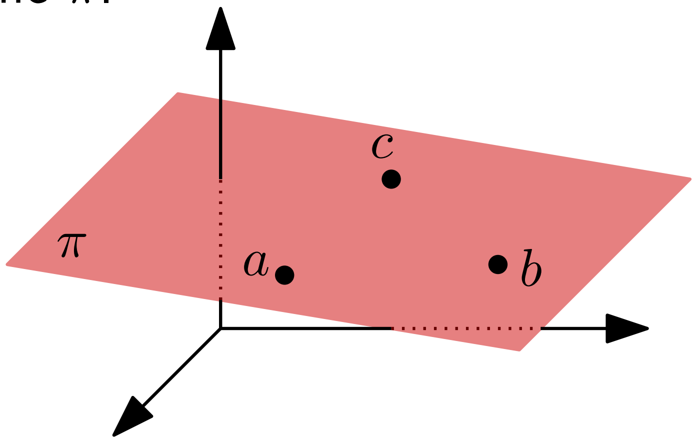
Relative position point-plane

If a, b, and c are not aligned, they define a plane π.



Relative position point-plane

If a, b, and c are not aligned, they define a plane π.



Point x lies in the halfspace $\pi^+ \iff det(x, a, b, c) > 0$.
Point x lies in the plane $\pi \iff det(x, a, b, c) = 0$.
Point x lies in the halfspace $\pi^- \iff det(x, a, b, c) < 0$.

2D application: Relative position point-circle

Proposition 1

The intersection of the paraboloid whose equation is $z = x^2 + y^2$ with a non vertical plane is a curve that projects orthogonally onto a circle in the plane $z = 0$.

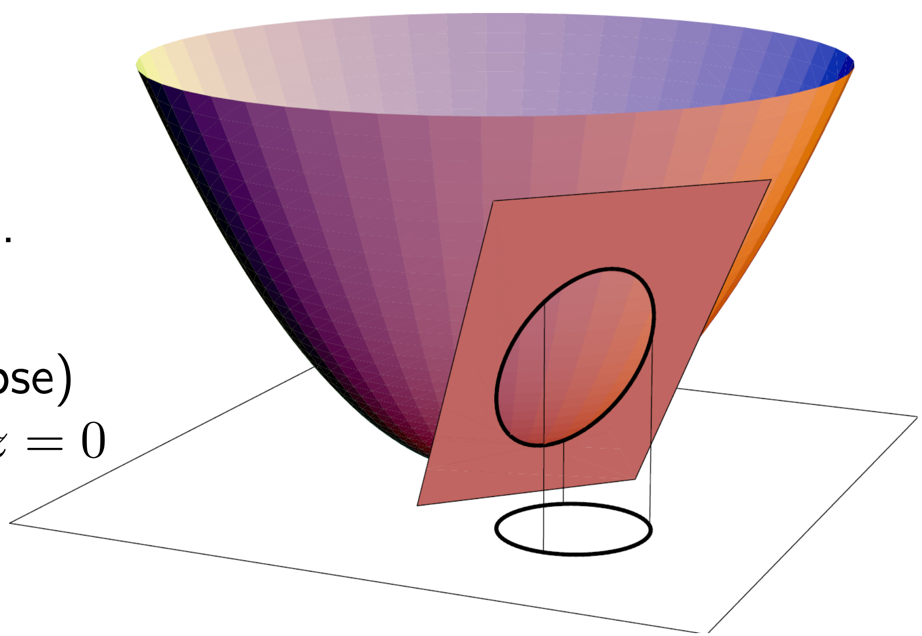
2D application: Relative position point-circle

Proposition 1
The intersection of the paraboloid whose equation is $z = x^2 + y^2$ with a non vertical plane is a curve that projects orthogonally onto a circle in the plane $z = 0$.

Proof:

$$\left.\begin{array}{l} z = x^2 + y^2 \\ z = 2ax + 2by + c \end{array}\right\} \quad \left.\begin{array}{l} x^2 + y^2 = 2ax + 2by + c \\ z = 0 \end{array}\right\} \quad \left.\begin{array}{l} (x-a)^2 + (y-b)^2 = c + a^2 + b^2 \\ z = 0 \end{array}\right\}$$

- If $c < -a^2 - b^2$,
  the intersection is empty.
- If $c = -a^2 - b^2$,
  the intersection is the point $(a, b, a^2 + b^2)$.
- If $c > -a^2 - b^2$,
  the intersection is a curve (in fact, an ellipse) that projects onto the circle of the plane $z = 0$ whose center is (a, b) and whose radius is $r = \sqrt{c + a^2 + b^2}$.

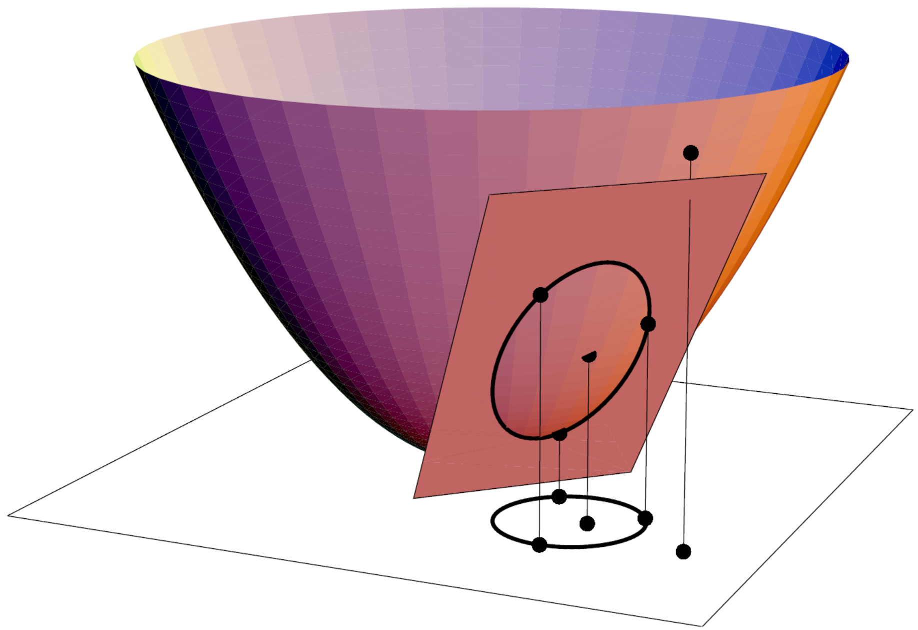


2D application: Relative position point-circle

Proposition 2

Let x, a, b, c be four points in the plane $z = 0$, and let x^*, a^*, b^*, c^* respectively be their vertical projections onto the paraboloid $z = x^2 + y^2$. If a, b, c are not aligned, let C be the circle through a, b, c, and let π be the plane through a^*, b^*, c^*.
Then:

- The point x lies in the circle C if and only if x^* lies in the plane π.
- The point x lies in the interior of the circle C if and only if x^* lies in the lower half-space determined by π.
- The point x lies in the exterior of the circle C if and only if x^* lies in the upper half-space determined by π.

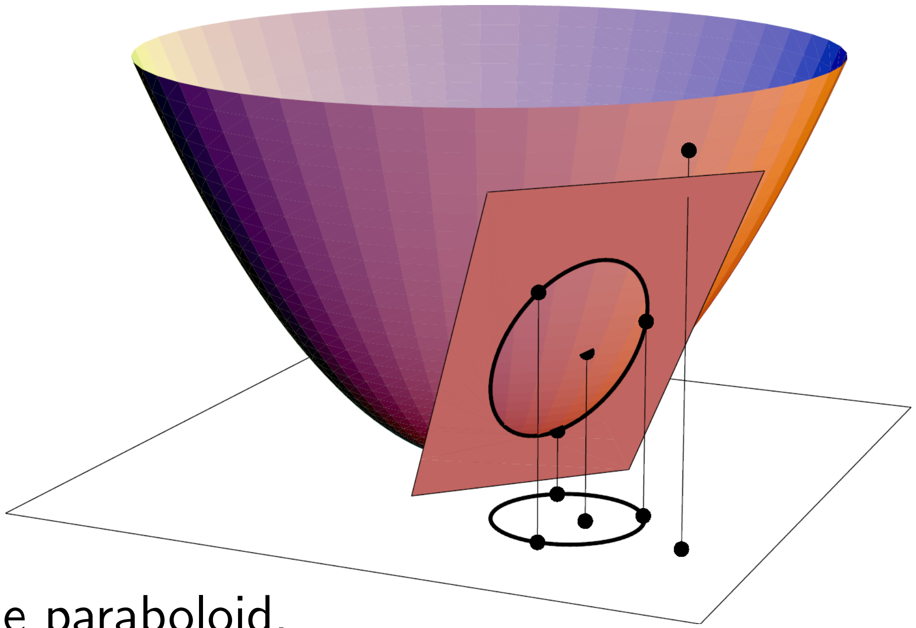


2D application: Relative position point-circle

Proposition 2

Let x, a, b, c be four points in the plane $z = 0$, and let x^*, a^*, b^*, c^* respectively be their vertical projections onto the paraboloid $z = x^2 + y^2$. If a, b, c are not aligned, let C be the circle through a, b, c, and let π be the plane through a^*, b^*, c^*.
Then:

- The point x lies in the circle C if and only if x^* lies in the plane π.
- The point x lies in the interior of the circle C if and only if x^* lies in the lower half-space determined by π.
- The point x lies in the exterior of the circle C if and only if x^* lies in the upper half-space determined by π.



Proof:

Due to Proposition 1 and the convexity of the paraboloid.

2D application: Relative position point-circle

Corollary

Let a, b and c be three non aligned points in the plane, that appear angularly sorted in counterclockwise order in the circle C that they determine.
Let x be any point in the plane.
Then:

- The point x lies in the circle C if and only if $\det(x^*, a^*, b^*, c^*) = 0$.
- The point x lies in the interior of C if and only if $\det(x^*, a^*, b^*, c^*) < 0$.
- The point x lies in the exterior of C if and only if $\det(x^*, a^*, b^*, c^*) > 0$.

2D application: Relative position point-circle

Observation
In order to compute the determinant of the previous corollary, it is convenient to do the calculations in terms of the differences between the values of the coordinates of the points involved, and to avoid making calculations (specially, products) in terms of the coordinate values, if possible:

$$\begin{vmatrix} x_1 & x_2 & x_1^2+x_2^2 & 1 \\ a_1 & a_2 & a_1^2+a_2^2 & 1 \\ b_1 & b_2 & b_1^2+b_2^2 & 1 \\ c_1 & c_2 & c_1^2+c_2^2 & 1 \end{vmatrix} = \begin{vmatrix} b_1-a_1 & b_2-a_2 & (b_1-a_1)(b_1+a_1)+(b_2-a_2)(b_2+a_2) \\ c_1-a_1 & c_2-a_2 & (c_1-a_1)(c_1+a_1)+(c_2-a_2)(c_2+a_2) \\ x_1-a_1 & x_2-a_2 & (x_1-a_1)(x_1+a_1)+(x_2-a_2)(x_2+a_2) \end{vmatrix}$$

FURTHER READING

J. O'Rourke
Computational Geometry in C
Cambridge University Press, 1994 (2nd ed. 1998), pp. 17-35.

F. P. Preparata and M. I. Shamos
Computational Geometry: An Introduction
Springer-Verlag, 1985, pp. 36-45.